

# System Engineering of Autonomous Space Vehicles

Michael D. Watson, Ph.D.,  
System Engineering Management Office  
NASA Marshall Space Flight Center  
Huntsville, Alabama  
[Michael.d.watson@nasa.gov](mailto:Michael.d.watson@nasa.gov)

Stephen B. Johnson, Ph.D.,  
Dependable System Technologies, LLC, and  
University of Colorado, Colorado Springs  
Colorado Springs, Colorado

Luis Trevino, Ph.D.,  
ISHM and Automation Branch  
Jacobs Technology  
Huntsville, Alabama

**Abstract—** Human exploration of the solar system requires fully autonomous systems when travelling more than 5 light minutes from Earth. This autonomy is necessary to manage a large, complex spacecraft with limited crew members and skills available. The communication latency requires the vehicle to deal with events with only limited crew interaction in most cases. The engineering of these systems requires an extensive knowledge of the spacecraft systems, information theory, and autonomous algorithm characteristics. The characteristics of the spacecraft systems must be matched with the autonomous algorithm characteristics to reliably monitor and control the system. This presents a large system engineering problem. Recent work on product-focused, elegant system engineering will be applied to this application, looking at the full autonomy stack, the matching of autonomous systems to spacecraft systems, and the integration of different types of algorithms. Each of these areas will be outlined and a general approach defined for system engineering to provide the optimal solution to the given application context.

**Keywords—**Autonomy, Goal-Function Tree, Integrated System Health Management, Spacecraft

## I. INTRODUCTION

Human missions beyond the Earth's planetary system will require autonomous systems to control the spacecraft. Communications latency will require near term responses to system and environmental conditions be handled onboard the spacecraft in collaboration with the flight crew. Once the vehicle has travelled 2 light minutes from Earth, tactical situations will be handled onboard as response time will take 5 minutes. While nominal communications have shown a tolerance to 5 minutes of communication delay (round trip)<sup>1</sup>, mission critical or crew safety issues will require more immediate responses. This latency will expand out to 20

minutes one way (41 minutes response time) as illustrated in Figure 1. While autonomous systems will be working in conjunction with the crew, the limited number of crew members available will require many functions to be handled by the autonomous systems and only mission or life-threatening situations involving direct crew interaction. The complexity of a human-rated spacecraft will amplify this situation, with life support, electrical power, thermal management, avionics, communication and tracking, structure, vehicle management, and propulsion all needing to be managed in an integrated fashion. In particular, the interactions among these systems and the environment can create many unique and unexpected situations to be managed.

There is limited data on fully autonomous operations of spacecraft and the experiments that have been conducted have mixed results in successfully controlling the spacecraft.<sup>2,3,4</sup> There have been various proposals to test individual applications on the International Space Station (ISS).<sup>5,6</sup>

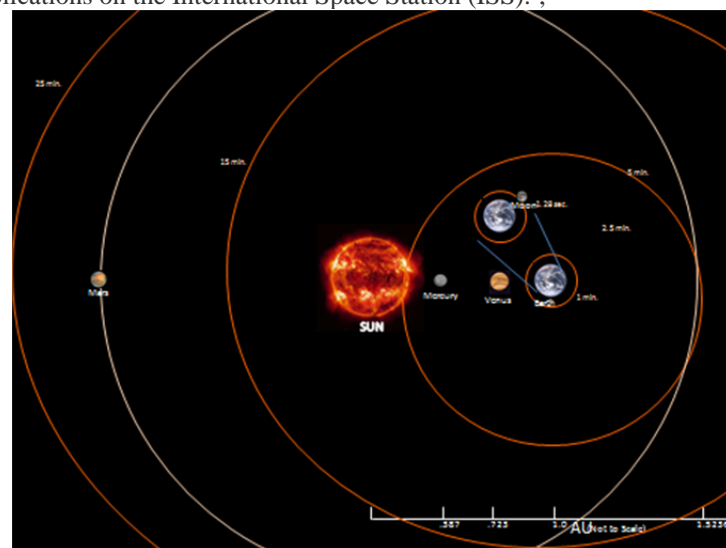


Figure 1: Communication Latencies from Earth

Engineering such a complex and inaccessible system (relative to the Earth) is a challenge. The system's interaction with itself and the environment has many complex effects. Understanding the physics in these situations is crucial to making decisions that maintain the vehicle integrity, crew safety, and mission success. Defining the state variables for the vehicle systems and their associated interaction effects provides a path to understand and properly manage the vehicle as a whole. State variables define the state of each physical and information system and provide a means to efficiently manage the system with a minimum set of measurements. With the state variables known, the specific autonomous algorithms can be matched to each system, keeping in mind the physics being managed and the strengths and weaknesses of each autonomous algorithm. This will optimize both the management of the system and the interaction among the various autonomous algorithms in managing the systems as a whole. This paper presents such an approach. We will briefly discuss state variables for a human-rated spacecraft in the Goal-Function-Tree context, and then give a short survey of various autonomous algorithms. The matching of the autonomous algorithms with the systems will then be discussed, defining a pathway to system engineering (SE) of human rated autonomous spacecraft.

## II. SPACECRAFT SYSTEMS

Figure 2 illustrates a potential Human-Rated Beyond Earth Orbit (BEO) Spacecraft. The basic systems necessary for a BEO human mission include: Propulsion, Structure, Thermal Management, Environmental Control and Life Support Systems (ECLSS), Electrical Power, Avionics, flight control system, Communication and Tracking, Vehicle Management (Guidance, Navigation and Control (GN&C) and Mission and Fault Management (M&FM)). Each of these systems has unique physics and responses to various interactions. A comprehensive review is not possible here. The following provide a short summary of each of these systems.



Figure 2: Possible Mars Exploration Vehicle

### A. Vehicle Management

Vehicle management consists of the functions that manage and control the vehicle as a whole. These include both GN&C, and M&FM. GN&C has long implemented automated control based on the Kalman filter.<sup>7</sup> The algorithmic approach fits very well with propagating the kinematic state of the vehicle to future states. M&FM provide the overall management of the vehicle systems, integrating the control loops across the vehicle and also provide management of failures as they are detected and diagnosed in flight. These

specific algorithms are tied to the physics of the other subsystems as discussed in this section. They also constitute the mission execution, vehicle control, and vehicle health management functions as discussed in the autonomy stack below.

### B. Flight Control System

For a spacecraft, flight control systems can be thrust vector control (TVC) systems, reaction control systems (RCS), or attitude control systems (ACS). TVC systems vector the thrust of the nozzle to affect the steering of the vehicle within a few degrees of the propulsion system center line. These systems can be hydraulic, electric, or hot gas driven. The physics in each of these systems differs greatly and the specific type will have unique control responses, operational lives, and vectoring force and accuracy. RCS and ACS<sup>8</sup> systems are similar, RCS usually indicating a system with larger force thrusters than an ACS. These systems can be either pressure fed or pump fed. Again, the physics in these systems vary greatly in control responses, operational lives, and thrust level. All of these systems are managed by the GN&C control algorithms in a closed loop fashion through the attitude sensors.

### C. Thermal Management

Thermal Management or Thermal Control systems (TCS) maintain the heat of the spacecraft including all systems and living areas within specified temperatures. The sources of heat across the spacecraft are many including propulsion system, electrical power system (especially if a nuclear reactor is used), ECLSS (which includes thermal management of living environments), avionics, and communications and tracking. Convection is not possible in micro gravity environments so heat transfer must be managed by conduction or radiation. Thus, heat from spacecraft systems and environments typically conduct heat into a transfer fluid which then passes through a radiator to radiate heat into space.<sup>9</sup> Because all spacecraft systems involve some level of heat generation, the thermal management system is highly coupled to other vehicle systems and must balance heat load and responses to variations across all systems and their operational requirements. Finite Element Models and Computational Fluid Dynamics are important to properly model these systems and the complex interactions they entail.

### D. Propulsion

Human-rated spacecraft must have reliable propulsion available through the life of the mission (typically 48 months<sup>10</sup>) and provide sufficient thrust profiles to maintain short planetary transfer times. The propulsion system includes the fuel and oxidizer storage systems, pumps to establish the mass flow rate, combustion chamber, and nozzle. Possible options for the propulsion system include chemical and nuclear thermal propulsion. The physics of these systems are again very different. Chemical propulsion systems have both

oxidizer and fuel mixing at high rates and temperatures to produce thrust following the rocket equation. The heat of combustion is the source of the gas acceleration through the nozzle exit. Nuclear thermal propulsion uses heat transfer from the reactor to a fuel (no oxidizer needed) to provide the acceleration of the gas. In a cryogenic chemical system, the storage of cryogenic propellants is a large concern. These propellants must be available until the mission is complete without boiling off (becoming gas) over the long mission profile. Nuclear thermal propulsion involves a reactor core that may also serve as a source of electrical power generation. Propellant management is easier as higher temperature liquids may be effectively heated by the reactor core. Cooling loops for the core may involve high boiling point liquids to transfer excess heat.

#### *E. Structure*

Structure includes both the static structure and the mechanisms necessary to turn and point solar arrays, thermal radiators, and antennas. The static structures are monitored by the GN&C for flex moments which affect trajectory control. These structures are fairly passive and if modeled need to be modeled by finite element models. The dynamic mechanisms are more complicated and need to be monitored for environmental degradation, wear on parts, and thermal deformations.<sup>11</sup>

#### *F. Electrical power*

Electrical power systems (EPS) encompass power generation, power storage, and power distribution and regulation. EPS can be implemented in a number of ways. Power can be generated by solar arrays, fuel cells, or nuclear reactors. Power is stored primarily by batteries although heat engines converting power to mechanical (rotational) energy are also possible. Power distribution includes regulation and voltage conversion. Distribution is generally through electrical wiring and power circuitry.<sup>12</sup> Optical transmission is also possible but currently has lower power efficiency. For a human-rated spacecraft it is expected that several types of power generation systems will be employed including possible solar arrays and nuclear reactors. The physics of each of the systems varies greatly.

#### *G. Avionics*

Avionics consist of the flight computers, data networks and busses, instrumentation (sensors, data conditioning, data acquisition systems), and software. These systems interact with every part of the vehicle and become the nervous systems for the vehicle management functions.<sup>13</sup> The physics involved with microelectronics, electromagnetic wave propagation, and various sensors are broad and complex. Avionics are sensitive to several environmental, design, and fabrication characteristics. Environments for avionics must be well managed including temperature, humidity (crew cabin), pressure (crew cabin or zero pressure), shock and vibration,

radiation, electrical, as well as dust and other contaminants. These must be controlled on the ground (during launch processing) and in space.<sup>14</sup>

#### *H. Communication and Tracking*

Communication systems include the transmitters, receivers, transmission lines, and antennas necessary to communicate with Earth, other ships (e.g., landers), satellites (e.g., planetary monitors or planetary communication satellites), and the planetary surface (i.e., landing party). These systems use a variety of signal transmission and encoding techniques that have varying susceptibility to noise sources and resulting bit error rates (BER).<sup>15</sup> Tracking includes both tracking of remote antennas and tracking of other spacecraft or debris (e.g., meteorites) that may threaten the vehicle. These systems couple directly with the GN&C algorithms to provide communication stability and hazard avoidance.

#### *I. ECLSS*

Environmental Control and Life Support Systems consists of several chemical and biological processes to manage air quality, waste and recycling, food storage and preparation, temperature and humidity, and fire detection and suppression.

### III. SPACECRAFT SYSTEM STATE VARIABLES

The state-based, goal-based system engineering method espoused in this paper specifies that a Goal-Function Tree (GFT) model<sup>16</sup> of the system should be constructed from the beginning of the SE process, and should be elaborated into further depth and detail as design choices are selected. The GFT provides a number of benefits, among which are a rigorous requirements (goal) definition and traceability in functional success space, beginning the development of fault trees by taking the logical complement of the GFT, analysis and definition of the required system health management and fault management to protect system goals, and most intriguingly for this paper, the creation of a physically and logically accurate tree structure that forms the starting point of the autonomous artificial intelligence for the system that can be used in system operations. While the SE and SHM/FM capabilities the GFT provides is important and useful, for a crewed Mars mission, the latter capability to provide the starting point of system autonomy development is essential.

Since the rigor and physical accuracy of the GFT depends on the comprehensive and systematic use of state variables to define goals and functions, defining the state variables associated with goals is a required step. While it is beyond the scope of this paper to attempt to define a detailed GFT for a crewed Mars mission, some basic insights as to what the GFT will look like and the kinds of state variables that it will use can be provided here. For a crewed Mars mission, as with many other kinds of systems and missions, it is useful to consider proposed mission phases and the goals associated with them, and then to determine the state variables that must be controlled within relevant ranges for these goals to be met.

In general for the GFT, there will be a unique tree structure for each major mission phase, which correspond to different ranges and values of state variables, and different state variables used during each phase.

A crewed Mars mission requires the transportation of the crew from Earth to the surface of Mars and back to Earth, and then the accomplishment of some specific scientific and technical goals while on the surface (and possibly during the journeys to and from Mars as well). This immediately implies three kinds of goals: transportation goals, crew health and safety goals, and scientific and technical goals.

At the top level, transportation goals can be stated rather quickly in terms of state variables, since position, velocity, and acceleration state variables completely define where the system must be at any given time within the mission, within specified bounds. To achieve these positions, velocities, and accelerations, the transportation system components will need to provide control of the vehicle's attitude and attitude rates to point the vehicle in the proper directions, thrust to provide acceleration, and some means to rotate the space vehicle(s) in space, whether through thrusters, reaction wheels, or some other mechanism. Provision of thrust requires control of propellant speeds and combustion (or accelerations), which in turn will entail state variables that must be controlled through chemical or electrical state variables (if using electric propulsion, for example). During entry, descent, and landing, the entry and descent vehicle may use aerodynamic forces instead of thrust, and if aerodynamic surfaces are used, then the control of these surfaces will entail their own state variables representing the movement of these surfaces. If a rover is to be used on the surface of Mars, then movement will require control of wheel spin rates and pointing directions. For each key mission phase, even if the same state variables of position, velocity, and acceleration are controlled, the values and ranges to which they are controlled vary at each phase, and the lower level state variables that produce the accelerations or rotations change. From the point of view of the GFT, for transportation goals, the top level of the trees for each phase often look quite similar, since the point of the transportation goals is to achieve certain positions, velocities, and accelerations. However, the lower levels of the trees for each phase will differ as the physics and means of controlling them change, such as with rocket engines using gimbals, thrusters, aerodynamic surfaces, and wheel rotations.

Like the transportation goals, the crew health and safety goals have strong similarities across mission phases. These too can be specified in terms of state variables. While the most direct measures of crew health relate to their individual life signs such as heart rate, respiration rate, food intake and defecation, water intake and urination, and the like, to achieve these the crew need breathable air, water, food, limited accelerations, controlled temperature ranges, time for activity and sleep, and so on. Each can be defined with state variables that can be readily defined: oxygen concentration, carbon dioxide concentration, atmospheric pressure, water and food mass, linear and rotational accelerations, air temperature, defecation and urination mass, etc. Since the needs of humans

in these basic senses are relatively constant, these do not change much during the mission, though the means to provide them may change based on having a pressurized living area versus space suits, and limiting accelerations on the launch vehicle, in space, during entry, descent, and landing (EDL), in a rover, in a Mars ascent vehicle, and for Earth entry. Potable water might be brought from Earth, or processed and captured in situ from Martian soil. Recycling or disposal of waste can also be defined in terms of mass and of chemical and physical state variables. Other things needed for human comfort and happiness are perhaps less easy to define in terms of state variables, such as entertainment and so on, but some might be, such as color schemes and volume of living quarters for each individual.

The purpose of the crewed Mars mission is to actually accomplish certain scientific and technical tasks, and to return the resulting information to Earth. Most of this information will be sent via communication systems, whose capacity is defined by the required transmission rates, which in turn are supported by radio frequency subsystem capabilities of radiated power, signal strength at various beam widths, and so on. Each of these scientific and technical tasks is generally describable in terms of the information to be gathered, with success often measured by the amount of data gathered, and of the relevant kind and quality. If the crew are unable to perform these tasks, then for each task not completed or only partially completed, this can be represented in state variable terms as losses to some fraction of the information that was specified as a goal for the mission. These goals could be altered during the mission, and likely will be as the crew finds it is able to accomplish more, less, or different goals than originally planned.

Finally, all of these goals are achieved by the use of specific hardware, software, and procedures. Each of these have their relevant sets of state variables appropriate to the functions they perform to achieve their subgoals. Thus computers of certain processing speed and memory perform functions, and are powered by electrical power generated by some means, whether solar, nuclear, or chemical, and then distributed through a power distribution system. All of these things are readily specified in terms of computing and electrical state variables. The software itself inherently manipulates internal and external state variables, the latter through sensors and actuators operated through various control systems. The control systems themselves can be specified in terms of the state variables they control, and which do the controlling. As the system is defined in the SE process, they can be laid out in a GFT form, which in turn forms the basis for autonomous algorithms that control the vehicles.

#### IV. AUTONOMY STACK

Autonomy algorithms have specific functions that interact in a defined manner.<sup>17</sup> These functions can be split at the full vehicle stack (Figure 3) and subsystem level (Figure 4). The autonomous functions necessary to manage a spacecraft include: integrated system health management (ISHM),<sup>18</sup>

system control, mission execution, mission planning, and a database of mission objectives and constraints (such as limits on responses to protect crew safety). In Figure 3, it can be seen that there are many management loops necessary to control the spacecraft. The inner-most loop is at the subsystem level and is broken more fully in Figure 4. The next loop is the vehicle management loop across all vehicle systems on the spacecraft. The mission execution loop involves the mission control functions such as guidance and navigation, control responses based on these updates, and adjustments to maintain mission objectives and constraints. The mission planning loop involves updates and changes to mission plans based on vehicle system status, mission objectives, and mission constraints. Mission planning can also result in revision of mission objectives and constraints with approval of the flight crew.

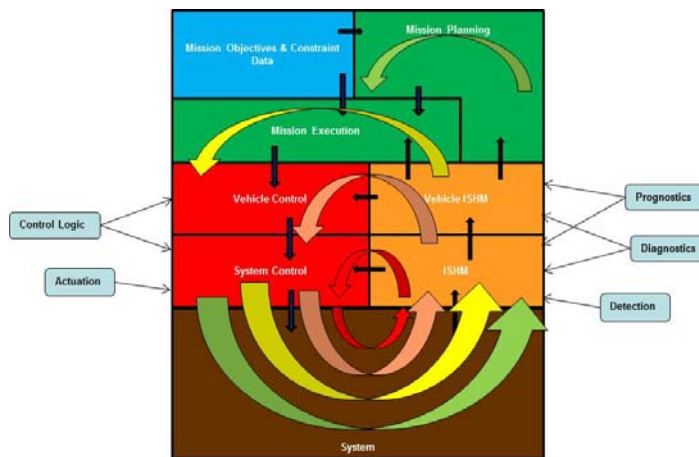


Figure 3: Autonomous System Stack

At the subsystem level, ISHM is broken out into its component functions (Figure 4): system monitoring, diagnostics, and prognostics. System monitoring includes the data acquisition system functions to measure and collect state variable data. This information is passed to both system management control loops and to the vehicle control loops to ensure vehicle management is done with a proper view of the actual system state. Diagnostics determine the actual system state based on the state variable measurements and their defined ranges for nominal and off nominal performance. Prognostics predict future system states including remaining useful life in system operations and consumables (as applicable). Vehicle control functions are included at the system level where performance is calculated based on measurements, and system control is decided based on system current performance, diagnostics, and prognostics. Vehicle control also uses this information to effect coordinated changes between systems, particularly when system interactions are driving internal system responses. This is important to ensure a system response to a change in state does not adversely create a conflict in the state of an interacting system.

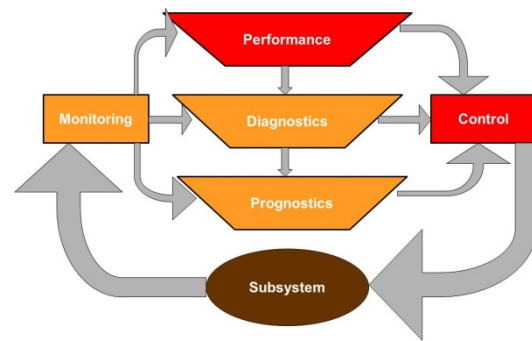


Figure 4: System Level Autonomous Management Loops

## V. CANDIDATE AUTONOMOUS ALGORITHMS FOR THE SPACECRAFT SYSTEMS

The sole intent of implementing autonomous algorithms is to reliably control the spacecraft without, or with limited human intervention. This includes the ability to respond to abnormal situations such as sensor failures or loss of communication links within the spacecraft. The principal autonomous algorithms being considered have already been investigated by the aerospace and academic community with focus on their diverse applications. The algorithms investigated include expert systems, neural networks, Bayesian belief networks, model based reasoning, and fuzzy logic. These methods and their hybrids have been demonstrated in marine, space, industrial, and aviation applications. Many of the algorithms can be used for diagnostics, prognostics, and planning applications.

### A. Expert System

Traditionally, an expert system is a computer program that simulates the judgment and behavior of a human or an organization that has expert knowledge and experience in a particular field. Typically an expert system contains a knowledge base containing accumulated experience and a set of rules for applying the knowledge base to each particular situation that is described to the software program (expert system). Sophisticated expert systems can be enhanced with additions to the knowledge base or to the set of rules. For an autonomous system, an expert system is viewed as the central authority in a distributed automatic and/or network of autonomous subsystems. Unlike its ground-based counterparts, for spacecraft, an onboard expert system will need to be hosted on an accommodating processing environment that could certainly impact the onboard avionics constraint parameters (e.g., thermal limits, weight, throughput, bandwidth). With the trending of current technologies, it is forecast that present day mission management (software) systems will incorporate more advanced algorithms and evolving into more "expert system like" systems. Verification and validation will be a challenge, but this is expected to be mitigated with the plethora of existing software engineering research on V&V for the presented algorithms (e.g., by NASA, Carnegie Mellon University, Software Engineering Institute).



## B. Neural Networks

Neural networks (NN) have been popular in a wide variety of applications, particularly for pattern recognition and case-based reasoning applications. NN are theoretically analogous to the way the human brain is structured and processes information. In regards to their development they are mathematically based on gradient descent methods with a variety of hardware and software implementation schemes. Issues in training and initial conditions are vital and largely dependent on the specific application. NN are ideal for control of highly nonlinear systems, interpolation and real time adaptation in the event of spacecraft subsystem reconfiguration due to unforeseen system disturbances or loss of a related subsystem. Configurations for their implementations have been demonstrated in aerospace applications, for example, for control of actuators or model-based applications in a variety of aviation applications such as dealing with a failed aileron and reconfiguring systems for degraded mode or fail safe applications and salvaging the aircraft, the mission, and protecting human life. A known application entails a neural flight control architecture based on an augmented model inversion controller.<sup>19</sup> This direct adaptive tracking controller integrated feedback linearization theory with both pre-trained and on-line learning neural networks. Neural networks can be viewed as quick response alternative control schemes for providing backup services or dealing with anomalous situations including using their pattern recognition strengths to isolate faults. Similar to other algorithms, they do have their issues such as extrapolation is often unreliable and accuracy of real-time learning (in unsupervised mode). For autonomous algorithm applications they are well suited for parallel processing. However, they may be very mathematically intense and can grow into a massive interconnection problem, where sometimes a global optimal solution may be difficult to reach. This can be mitigated with reasonably relaxed training constraint parameters and methodical design for the target application.

## C. Fuzzy Logic

Another technology applicable to autonomous systems is fuzzy logic-based systems, largely based on classic mathematical set theory and analogous to neural networks. Like neural networks, fuzzy systems have been demonstrated in a variety of complex nonlinear aerospace and commercial applications. A priori knowledge of the subsystem is a necessity for training and development of fuzzy logic-based systems. Implementation is typically much simpler due to the relative ease of software implementation and/or hardware (such as fuzzy chips) as evident by their utility in applications such as control of household appliances, cameras, locomotive braking systems, and aerospace systems. Fuzzy systems are very analogous to a proportional-integral-derivative (PID) controller and have been demonstrated to be much more robust than classic controllers and dealing with uncertainty is inherent in their makeup. For spacecraft autonomy, fuzzy logic algorithms are prime candidates for their application to various

subsystems such as augmenting control in caution and warning scenarios with possible degraded mode operations, ideal for control of TVC or backup systems (in the loop), local subsystem onboard control (e.g., separate health node to guide critical systems, like solid rocket boosters, to safety during a separation mode and can be considered for steering of spacecraft in uncertain states).

## D. Model-Based Reasoner

A model-based reasoner (MBR) can take many forms depending on the application. For autonomous systems MBR have been demonstrated in actual space applications such as the NASA Ames Research Center-developed Livingston L2 engine concept for the Path Finder mission.<sup>20,21</sup> Models are developed using extensive domain knowledge and need to be implemented in a well-crafted software architecture using an efficient programming language and operating system capable of dealing with conflict resolution, efficient processing, and avoiding common issues in software processing in embedded systems for mission critical applications (i.e., software health management). There are numerous MBR applications for other applications in which to leverage/investigate other features to map to analogous spacecraft functions. Furthermore, MBR algorithms can be implemented as serving as the key reasoning sub-element in a mission manager system leading to a hybrid type of expert system. Principally, MBR algorithms are ideal for vehicle diagnostics for comparing expected subsystem behavior with actual behavior (analogous to a Kalman Filter providing model information a central control scheme). Diagnostics using MBR methods have been proven to provide very fast and reliable response. Like other autonomous algorithms, there are disadvantages such as V&V and inference using the latest reliable information. In general the biggest hurdle for this and all advanced algorithms for autonomous systems is technology infusion, an ongoing topic in prior and current forums such as those for prognostics, diagnostics, V&V, and related software engineering and aerospace industries.

## E. Bayesian Belief Networks

Bayesian belief networks (BBN) have also been extensively applied in a variety of diverse applications and are ideal for supporting the credibility of the state(s) in a given system(s). The central method composed in the BBN is Bayes's rule being used in a cook book script method for propagating information in order to assess the qualitative state of a system and/or its subsystems. The method basically entails prior and likelihood beliefs to propagate throughout a network of system state nodes represented as state variables. These nodes then statistically ascertain the state of the subsystem represented by the associated state variable. Again, BBN are heavily dependent on a priori system knowledge and their implementation needs to be carefully integrated as a passive system providing credible information to an existing expert system or other like central authority. BBNs have been extensively applied in aerospace systems such as air breathing

jet systems and they've also been effective as sensor data qualification systems.<sup>22</sup>

The presented autonomous algorithms above will each have memory and processing needs requiring a commensurate processing architecture. This architecture will need to be flexible and extensible to accommodate mission plans, scalable, and be configurable. The architecture will also need to support the processing and management of data. Learning will be a key topic for each of the proposed autonomous algorithms. With knowledge of each of the spacecraft subsystems, the learning algorithms can be determined and depending on the complexity of the subsystem or mission phase, training may be able to be performed unsupervised, otherwise, will need to be addressed in delayed human-in-the-loop response. The principal aim of any spacecraft autonomous algorithm is to manage vehicle functions and subsystems to reliably guide the spacecraft, whether it's an expert system for the whole spacecraft or a single autonomous subsystem (such as an engine controller). The spacecraft central computer and/or its vehicle's subsystems will especially need to have capabilities to reliably reason on known and unforeseen failure scenarios. As stated above, technology infusion will be successful once trust of the algorithms is proven in flight like (test bed) environments and supported with a high fidelity flight avionics computing architecture.

## VI. AUTONOMOUS ALGORITHM INTEGRATION

The complexity of the integration of the autonomous algorithms is many faceted. As presented above there are three main aspects to integrating the algorithms with the vehicle systems and with each other: System level management, vehicle level management, execution and planning. At the system level, the key is understanding the physics of the system and selecting an autonomous algorithm that can effectively (take the necessary actions based on all interactions) and responsively (take the necessary action in a timely manner) manage the physics. These physics are driven by the internal system processes, interactions with other systems, and interactions with the environment, all of which must be managed by the algorithm. At the vehicle level, the focus is on integration of the systems into a cohesive and response management system. The physics effects on the vehicle at this level are essential to taking proper responses to planned and unplanned conditions. The interactions between systems are managed to ensure systems respond cooperatively, not competitively, such that systems do not counter each other's actions leaving the vehicle in a failed state. The mission execution function mitigates these affects through adjustment to system control parameters in response to specific physical events. Mission planning involves the proper knowledge of the current vehicle states, the progress toward specific mission objectives, and re-planning (with crew approval) to ensure future vehicle states will stay within mission objectives and constraints. Note that the Earth-based controls will also be involved with the strategic mission

aspects of re-planning as well as the crew for the tactical mission aspects of re-planning.

System-level algorithm matching involves knowledge of the system transfer functions which include external system and environment interactions. These algorithms will be controlling the system responses, hence, control theory is important in implementation. The physics will define the poles and zeros of the control system and the relative proximity of the system response to these locations. Essential in this, is the particular transfer function. These functions must be defined and matched with the characteristics of the autonomous algorithms. Expert Systems, Neural Networks, Fuzzy Logic, and Model-Based Reasoners are all candidates for spacecraft systems.

Following the discussion of the spacecraft systems above, vehicle management GN&C algorithms are well matched to the fuzzy logic Kalman Filter. The ability of the filter logic to consider both the current and future states well adapts this algorithm to the physics of the vehicle. M&FM algorithms are directly coupled to the systems and must be matched to the specific physics of each system. Flight control systems are a direct application of control theory and the autonomy management functions must incorporate these aspects. These systems require quick responses in operation, so the algorithm will need to support this. Neural Networks or Expert Systems are candidates to provide the autonomous control in these cases. Thermal Management Systems interconnect through all the vehicle systems. The autonomy will need to be well suited for determining the impacts of these interactions, locating the sources of unexpected perturbations, and projecting the impact of mitigations in one area across all systems. Thermal responses are slow in propagation relative to other effects. Thus, Model Based Reasoners are good candidates for management of this type of system. Propulsion systems entail both slow, long term effects (such as fluid management and leaks) with non-linear quick reactions during engine firing. Thus, a combination of neural networks and expert systems or possibly Fuzzy Logic would be necessary to manage these systems. Structure and mechanisms are somewhat static as compared to the more dynamic engine or control thruster operations. Finite elements models are typically used to accurately design these systems for stress, strain, fracture control, loads, flexure, etc. If this level of modeling is required, a fuzzy logic or Bayesian belief algorithm may provide the best application to be able to consider aggregate points without executing detailed finite element analysis in response to anomalies. If this level of modeling is not necessary on board, fuzzy logic, Bayesian belief, or an expert system would provide a good approach. Electrical Power Systems and Avionics are highly interconnected. Management of bandwidth on shared resources is critical. This becomes acute if failures segment portions of the system. The architectures will need to be robust to handle these consequences, and so the autonomy will need to be able to handle the architecture and responses to failures. Model-Based Reasoners are well suited to make these kinds of adaptive control applications, where the model can be adapted to match

the architectural changes as necessary in flight. Communications applications are similar with bandwidth and location of remote transceivers essential. Power management is also critical to signal to noise ratios in communication. Model-Based Reasoners then provide a good solution for the management of communication. Tracking involves defining relative states between the vehicle and other external objects. Thus fuzzy logic Kalman Filters, similar to GN&C, provide a clear choice for tracking system management. ECLSS includes many chemical and biological aspects that must be managed. Crew safety and health constraints are essential to the operation of ECLSS. Thus, expert systems may provide the best approach to autonomy for ECLSS applications.

At the vehicle level, the integration aspects are essential. A full physical understanding resides at the vehicle level. Algorithms at this level look at the current state of the vehicle, interaction responses between the systems and environments, the prognosticated state of the vehicle, mission objectives, and mission constraints to manage the total vehicle execution of the mission. Thus, algorithms that can handle both the physical understanding and future trends is essential. Model-Based Reasoners or Expert Systems are both candidates for this type of autonomy. Each can handle the physical equations and also the prognostics of future states along the current mission plan. The Goal-Function Tree described above, which was developed during the SE process, can be used as one basis for the Expert System tree structures needed for vehicle-level reasoning, and as described in the next paragraph, planning.

Planning requires a much more focused view on potential future states of the vehicle given the current state and path to the future state. The algorithm must deal flexibly with mission objectives, as some failure states, may lead to an abort of certain mission objectives. System constraints will need to be maintained, though the crew can relax these if vehicle failures lead to a differing operating mode. In considering these aspects, Bayesian Belief Networks have the basis to execute mission planning tasks.

As can be seen, autonomous systems will not be a single algorithm but multiple algorithms, each matched to the specific system or vehicle function it is performing. The integration of these functions is an area of future work in spacecraft autonomy. Applications looking at autonomous system cooperation will be essential to the development of human rated spacecraft operated away from the Earth planetary system.

## VII. SUMMARY

Human exploration outside of the Earth planetary system (beyond Earth orbit) requires autonomous operation of the vehicle to deal with communication latencies, crew size limits, and vehicle complexity. A fully autonomous vehicle of this complexity will require multiple autonomous algorithms working cooperatively within a set of mission objectives and system constraints. The understanding of the physics of the systems, system interactions, and environmental interactions is essential to the system engineering of this complex system. The Goal-Function Tree methodology provides a system

engineering approach to define the vehicle state variables and their interactions. Using these state variables and the GFT structured hierarchy, among other resources, specific autonomous algorithms can be chosen based on their ability to properly handle the system physics. Algorithms at the vehicle level will also need to handle future projected states to enable safe mission execution and planning. Verification and validation approaches will need to be defined for these algorithms, both individually and as an integrated set. The integration responses of these algorithms are essential to a successful human mission and will require further study, development, and evaluation.

## REFERENCES

- <sup>1</sup> Frank, J., Spirkovska, L. ; McCann, R. ; Wang, L. ; Pohlkamp, K. ; Morin, L. , "Autonomous Mission Operations", IEEE Aerospace Conference, 2013, pp 1-20.
- <sup>2</sup> Pell, Barney ; Sawyer, Scott R. ; Muscettola, N. ; Smith, Benjamin ; Bernard, Douglas E., "Mission Operations with an Autonomous Agent", IEEE Aerospace Conference, Vol. 2, 1998, pp. 289 – 313.
- <sup>3</sup> Aljabri, Abdullah S. ; Bernard, Douglas E. ; Dvorak, Daniel L. ; Man, Guy K. ; Pell, Barney ; Starbird, Thomas W., "Infusion of Autonomy Technology into Space Missions: DS1 Lessons Learned", IEEE Aerospace Conference, Vol. 2, 1998, pp. 315 - 329.
- <sup>4</sup> Katz, Jacob G. ; Saenz-Otero, Alvar ; Miller, David W., "Development and Demonstration of an Autonomous Collision Avoidance Algorithm aboard the ISS", IEEE Aerospace Conference, 2011, pp. 1-6.
- <sup>5</sup> Haddock, Angie T. ; Stetson, Howard K., "Automated Operations Development for Advanced Exploration Systems", 2012 AIAA Space Operations.
- <sup>6</sup> Stetson, Howard K. ; Watson, Michael D. ; Shaughnessy, Ray, "HAL 9000 Space Operating System Real Time Planning Engines, 2012 AIAA Space Operations.
- <sup>7</sup> Wittman, Klaus, "Fundamentals" in Handbook of Space Technology, ed. By Ley, Wilfried ; Wittmann, Klaus ; Hallmann, Willi, AIAA, Virginia, 2009, pp 73-74.
- <sup>8</sup> Arbinger, Christian ; Lubke-Ossenbeck, Bernard, "Attitude Control" in Handbook of Space Technology, ed. By Ley, Wilfried ; Wittmann, Klaus ; Hallmann, Willi, AIAA, Virginia, 2009, pp 332 – 341.
- <sup>9</sup> Kohlase, Andreas ; Schlitt, Reinhard,, "Thermal Control" in Handbook of Space Technology, ed. By Ley, Wilfried ; Wittmann, Klaus ; Hallmann, Willi, AIAA, Virginia, 2009, pp 268-300
- <sup>10</sup> Watson, Michael D., "Photonics on the Mission to Mars", Proc. SPIE 8876, Nanophotonics and Macrophotonics for Space Environments VII, 2013, pp. 1 - 13.
- <sup>11</sup> Block, Joachim ; Kreuser, Johannes, "Structure and Mechanisms" in Handbook of Space Technology, ed. By Ley,



---

Wilfried ; Wittmann, Klaus ; Hallmann, Willi, AIAA, Virginia, 2009, pp 203-236.

<sup>12</sup> Ley, Wilfried ; Roder, Reinhard, "Electrical Power Supply" in Handbook of Space Technology, ed. By Ley, Wilfried ; Wittmann, Klaus ; Hallmann, Willi, AIAA, Virginia, 2009, pp 236-268.

<sup>13</sup> Montenegro, Sergio, "Data Management" in Handbook of Space Technology, ed. By Ley, Wilfried ; Wittmann, Klaus ; Hallmann, Willi, AIAA, Virginia, 2009, pp 361 – 380.

<sup>14</sup> Watson, Michael D. ; Varnavas, Kosta ; Patrick, Clint ; Hodge, Ron ; Byington, Carl S. ; Chau, Savio ; Baroth, Edmond C., "Avionics Health Management", in Stephen B. Johnson, Thomas J. Gormley, Seth S. Kessler, Charles Mott, Ann Patterson-Hine, Karl M. Reichard, Philip A. Scandura, Jr., eds. System Health Management: with Aerospace Applications (Chichester, United Kingdom: John Wiley United Kingdom, July 2011), Chapter 28, pp. 445 - 465

<sup>15</sup> Turner, Peter, "Communication" in Handbook of Space Technology, ed. By Ley, Wilfried ; Wittmann, Klaus ; Hallmann, Willi, AIAA, Virginia, 2009, pp 380 – 398.

<sup>16</sup> Johnson, Stephen B., Goal-Function Tree Modeling for Systems Engineering and Fault Management. AIAA Infotech@Aerospace (I@A) Conference, 2013, Boston, MA. August 19-22. AIAA Paper 2013-4576

<sup>17</sup> Watson, Michael D., Johnson, Stephen B., "A Theory of Vehicle Management Systems", IEEE Aerospace Conference, 2007, pp. 1-8.

<sup>18</sup> Johnson, Stephen B., "The Theory of System Health Management", in Stephen B. Johnson, Thomas J. Gormley, Seth S. Kessler, Charles Mott, Ann Patterson-Hine, Karl M. Reichard, Philip A. Scandura, Jr., eds. System Health Management: with Aerospace Applications (Chichester, United Kingdom: John Wiley United Kingdom, July 2011), Chapter 1, pp. 3-27

<sup>19</sup> R. Rysdyk & A. Calise, "Fault Tolerant Flight Control via Adaptive Neural Network Augmentation", AIAA 98-4483, August 1998

<sup>20</sup> J. Kaneshige & K. Gundy-Burlet, " Integrated Neural Flight and Propulsion Control System," AIAA-2001-4386

<sup>21</sup> J. Kurien et al, "Model-based Autonomy for Robust Mars Operations," NASA Ames Research Center technical internal report

<sup>22</sup> L. Trevino, C. Karr, et al, "SR-30 Turbojet Engine Real-time Sensor Health Monitoring using Neural Networks, and Bayesian Belief Networks," Applied Intelligence Journal, DOI 10.1007/s10489-006-0017-z, Springer Science + Business Media, LLC 2006